

# Banco de Dados

---

Modelo Relacional  
– Normalização –

Caetano Traina Júnior

# Normalização

---

## Controlando Consistência

- O controle de consistência pode ser exercido:
  - ▶ pelos gerenciador,
  - ▶ pelo aplicativos, ou
  - ▶ pelo própria construção do sistema.
- O controle obtido pela própria construção do sistema é, em geral, a melhor opção, pois não incorre em perda de eficiência durante a execução.

# Normalização

---

## Controlando Consistência

- O Modelo Relacional, devido à sua natureza inerentemente formal, dispõe de ferramentas conceituais que permitem modelar diversas formas de controle de consistência.
- O controle de através da própria construção do sistema é obtida no Modelo Relacional construindo-se as relações segundo regras que garantem a manutenção de determinadas propriedades.
- Às relações que atendem a um determinado conjunto de regras diz-se estarem em uma determinada **Forma Normal**.

# Normalização

---

## Controlando Consistência - Normalização de Relações

- O processo de normalização permite ao programador controlar quanto da consistência é garantida pela maneira de construção do sistema, e quanto deve ser responsabilidade dos aplicativos e/ou do SGBD.
- Normalizar demais diminui a eficiência dos aplicativos, e de menos abre flancos para inconsistências.

# Normalização

---

## Controlando Consistência - Normalização de Relações

### ■ Resumindo:

- ▶ A normalização de relações é a maneira de controlar a consistência das relações através de uma ferramenta conceitual;
- ▶ Uma relação que atende a determinadas propriedades é dita estar numa determinada Forma Normal ;
- ▶ Portanto, cada Forma Normal atende aos requisitos de uma determina forma de consistência de uma relação.

# Normalização

## Controlando Consistência - Normalização de Relações

### ■ 1ª Forma Normal:

- ▶ As formas normais têm nomes pelas quais são conhecidas.
- ▶ Para o modelo relacional, a forma normal mais importante é aquela chamada 1ª Forma Normal.
- ▶ Uma relação está na 1ª Forma Normal quando todos os seus atributos são Atômicos e Monovalorados.
- ▶ Atributos Atômicos: É um atributo que não é tratado em partes separadas em nenhuma outra relação da base de dados.
- ▶ Atributos Monovalorados: É um atributo que tem apenas um valor (não uma lista).

# Normalização

## Atributos Atômicos

### ■ Atenção:

- ▶ É um atributo do qual não ocorrem partes em nenhuma outra relação da base de dados.

### ■ Portanto:

- ▶ Não importa se o Domínio desse atributo não precisa ser a composição de dois ou mais domínios elementares nessa relação. O que importa é que em nenhuma relação da base de dados esse domínio é composto.

Aluno = {Nome, Idade, DataNasc., DataMatricula}

Turma = {Sigla, Número, Semestre, Ano, N#Alunos}

# Normalização

## Atributos Atômicos

Data de nascimento e de matrícula não precisam que seu domínio seja composto na Relação Aluno.

Aluno = {Nome, Idade, DataNasc., DataMatricula}  
Turma = {Sigla, Número, Semestre, Ano, N#Alunos}

Mas, do mesmo domínio “data” é usada a componente “Ano” em outra relação.



# Normalização

## Atributos Atômicos

Data de nascimento e de matrícula não precisam que seu domínio seja composto na Relação Aluno.

Aluno = {Nome, Idade, DataNasc., DataMatricula}  
Turma = {Sigla, Número, Semestre, Ano, N#Alunos}

Mas, do mesmo domínio “data” é usada a componente “Ano” em outra relação.

# Normalização

---

## Atributos Atômicos

Se não houvesse a relação Turma, ou não houvesse nela o atributo “*ano*”, então *DataNasc* e *DataMatricula* poderiam ser atômicos. Havendo essa relação, esses dois atributos não são atômicos.

# Normalização

---

## Atributos Monovalorados

- Ao contrário do atributo Atômico, um atributo ser ou não multivalorado é avaliado apenas na relação onde ele ocorre; não depende das demais relações da base.

# Normalização

## Como Normalizar Relações para a 1<sup>a</sup>. Forma Normal

- Uma relação que não esteja na 1<sup>a</sup>. Forma Normal pode ser colocada nela. A ação a ser tomada depende da causa da anomalia.
- Atributos não atômicos:
  - Cada um dos atributos não atômicos da relação são “quebrados” em seus atributos componentes.

~~Turma = {SiglaNumero, Sala, NHoras}~~

Turma = {Sigla, Numero, Sala, NHoras}

# Normalização

---

## Como Normalizar Relações para a 1<sup>a</sup> Forma Normal

### ■ Atributos Multi-valorados

- ▶ Existem duas maneiras de normalizar relações que contêm atributos multivalorados:
  - A quantidade de valores é pequena e conhecida a priori;
  - A quantidade de valores é muito variável, desconhecida, ou grande.

# Normalização

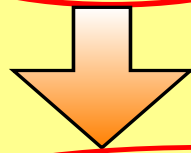
## Como Normalizar Relações para a 1ª Forma Normal

### ■ Atributos Multi-valorados

- ▶ Existem duas maneiras de normalizar relações que contêm atributos multi-valorados:
  - A quantidade de valores é pequena e conhecida a priori;

Substitui-se o atributo multivalorado por um conjunto de atributos de mesmo domínio, cada um monovalorado representando uma ocorrência do valor

Paciente = {Nome, GrausLentes, TipoSanguíneo}



Paciente = {Nome, GrauEsq., GrauDir., TipoSanguíneo}

# Normalização

## Como Normalizar Relações para a 1ª Forma Normal

### ■ Atributos Multi-valorados

- ▶ Existem duas maneiras de normalizar relações que contêm atributos multi-valorados:
  - A quantidade de valores é pequena e conhecida a priori;
  - A quantidade de valores é muito variável, desconhecida, ou grande.

Retira-se da relação o atributo multivalorado, e cria-se uma nova relação que tem o mesmo conjunto de atributos chave, mais o atributo multivalorado também como chave, mas transformado em monovalorado.

Aluno = {Nome, Idade, Disciplinas, NUSP}

Aluno = {Nome, Idade, NUSP}

Disciplinas = {Nome, Disciplinas}

# Normalização

---

## Controlando Consistência

- A 1<sup>a</sup>. Forma Normal é uma das maneiras de controlar-se consistência através da própria estrutura do sistema.
- A 1<sup>a</sup>. Forma Normal é fundamental para a própria conceituação do Modelo Relacional, e é exigida por todos os SGBDRs comuns.
- SGBDRs e a teoria correspondente que não exigem a 1<sup>a</sup>. Forma Normal são designadas como NF<sup>2</sup> (Non First Normal Form).
- Relações que não estão em 1<sup>a</sup>. Forma Normal são chamadas “*Nested Relations*” (Relações dentro de Relações).



# Normalização

---

## Controlando Consistência

A 1<sup>a</sup>. FN é uma das  
maneiras de se Controlar  
Consistência.

# Normalização

---

## Dependências Funcionais

- Uma outra maneira de controlar consistência é usando as dependências funcionais existentes entre os atributos armazenados.
- Dependência funcional é um conceito que baseia-se no reconhecimento que os valores de alguns atributos podem ser determinados a partir de outros. Esse conhecimento não pode ser inferido pelo SGBD, e portanto deve ser avaliado e informada pelo programador ao Banco de Dados.

# Normalização

## Dependências Funcionais

- Se o valor de um conjunto de atributos  $A$  permite descobrir o valor de um outro conjunto  $B$ , dizemos que  $A$  determina funcionalmente  $B$  ou que  $B$  depende de  $A$  e denotamos:

$$A \rightarrow B$$

NUSP  $\rightarrow$  Nome, Idade, Curso

Sigla, Sala, Hora  $\rightarrow$  CódigoTurma

Sigla  $\rightarrow$  Nome, Disciplina, NCréditos

# Normalização

## Dependências Funcionais


- As Dependências Funcionais deve ser identificadas pelo projetista do sistema sendo desenvolvido: não se pode “inferir” as dependências a partir da descrição da base.
- Dada uma instância de uma base qualquer, é possível verificar se dois conjuntos de atributos quaisquer atendem à uma Dependências Funcionais.

Para verificar se  $C_1 = \{A_1, \dots, A_g\} \rightarrow C_2 = \{A_n, \dots, A_k\}$  numa relação  $R$ , é necessário que para toda  $T_1(C_1) = T_2(C_1)$  também vale  $T_1(C_2) = T_2(C_2)$ .

# Normalização

## Dependências Funcionais

Alunos = {Nome, Curso, Idade}  
{<Mario, Compt., 21>,  
<Paulo, Eletr., 22>,  
<Almir, Fisio., 22>,  
<Marta, Comp., 21>,  
<Vânia, Eletr., 22>}

Nome  Curso ?  
Idade •


Sempre que o Nome se repetir, Curso ou Idade se repete?

# Normalização

## Dependências Funcionais

Alunos = {Nome, Curso, Idade}  
{<Mario, Compt., 21>,  
<Paulo, Eletr., 22>,  
<Almir, Fisio., 22>,  
<Marta, Comp., 21>,  
<Vânia, Eletr., 22>}

OK!!!

Nome  Curso ?  
Idade •

Sempre que o Nome se repetir, Curso ou Idade se repete?

# Normalização

## Dependências Funcionais

Alunos = {Nome, Curso, Idade}  
{<Mario, Compt., 21>,  
<Paulo, Eletr., 22>,  
<Almir, Fisio., 22>,  
<Marta, Comp., 21>,  
<Vânia, Eletr., 22>}

Curso  $\longrightarrow$  Idade ?

compt.  $\Rightarrow$  21

eletr.  $\Rightarrow$  22

fisio.  $\Rightarrow$  22

# Normalização

## Dependências Funcionais

Alunos = {Nome, Curso, Idade}  
{<Mario, Compt., 21>,  
<Paulo, Eletr., 22>,  
<Almir, Fisio., 22>,  
<Marta, Comp., 21>,  
<Vânia, Eletr., 22>}

OK!!

Curso  $\longrightarrow$  Idade ?

compt.  $\Rightarrow$  21

eletr.  $\Rightarrow$  22

fisio.  $\Rightarrow$  22



# Normalização

## Dependências Funcionais

Alunos = {Nome, Curso, Idade}  
{<Mario, Compt., 21>,  
<Paulo, Eletr., 22>,  
<Almir, Fisio., 22>,  
<Marta, Comp., 21>,  
<Vânia, Eletr., 22>}

Idade  $\longrightarrow$  Curso ?

21  $\Rightarrow$  compt.  
22  $\Rightarrow$  eletr., fisio.

# Normalização

## Dependências Funcionais

Alunos = {Nome, Curso, Idade}  
{<Mario, Compt., 21>,  
<Paulo, Eletr., 22>,  
<Almir, Fisio., 22>,  
<Marta, Comp., 21>,  
<Vânia, Eletr., 22>}

Não!!

Idade  $\longrightarrow$  Curso ?

21  $\Rightarrow$  compt.  
22  $\Rightarrow$  eletr., fisio.

# Normalização

---

## Dependências Funcionais

As Dependências Funcionais são informações semânticas fornecidas pelo projetista.

Não podem ser obtidas “olhando-se” um estado particular da base.

Dependências Funcionais fazem parte da base intencional, refletem a intenção do projetista.

# Normalização

---

## Dependências Funcionais

- Para controlar consistência, é necessário conhecer todas as dependências funcionais que existam.
- Sendo uma informação semântica, todas deveriam ser fornecidas pelo projetista.
- No entanto, é possível inferir-se dependências funcionais a partir de outras já conhecidas.
- A partir de um conjunto básico, é possível ampliar-se esse conjunto através de um conjunto de 6 regras de inferência.
- Note-se que não se pode inferir todas, apenas aquelas que podem ser obtidas através desse conjunto de regras.
- Assim, elaborar-se um conjunto inicial o mais completo possível é de importância fundamental para se controlar adequadamente a consistência.

# Normalização

## Dependências Funcionais

- As 6 regras de derivação de dependências funcionais são as seguintes:
  - ▶ Reflexiva: Se  $A \subset B \Rightarrow B \rightarrow A$
  - ▶ Aumentativa: Se  $A \rightarrow B \Rightarrow AC \rightarrow B$
  - ▶ Separação (Projeção): Se  $A \rightarrow BC \Rightarrow A \rightarrow B, A \rightarrow C$
  - ▶ Aditiva: Se  $A \rightarrow B, A \rightarrow C \Rightarrow A \rightarrow BC$
  - ▶ Transitiva: Se  $A \rightarrow B, B \rightarrow C \Rightarrow A \rightarrow C$
  - ▶ Pseudo-Transitiva: Se  $A \rightarrow B, BC \rightarrow D \Rightarrow AC \rightarrow D$

# Normalização

---

## Dependências Funcionais

- Essas 6 regras podem ser usadas para, a partir de um conjunto inicial de DF, obterem-se outras DF derivadas.
- No entanto, teoricamente apenas as regras Reflexiva, Aumentativa e Transitiva são suficientes, pois as demais podem ser obtidas pela aplicação sucessiva dessas 3.
- Esse conjunto é chamado completo e suficiente para inferir todas as DF a partir de um conjunto inicial.

# Normalização

---

## Normalização das Relações

- Existem diversas maneiras para se normalizar relações baseadas nas dependências funcionais. Elas garantem por construção que diversas inconsistências não têm como ocorrer nas relações assim normalizadas.

# Normalização

## Normalização das Relações

- Uma Relação está na 2<sup>a</sup>. Forma Normal quando:
  - ▶ Está na 1<sup>a</sup>. Forma Normal;
  - ▶ Todos os atributos que não participam da chave primária são dependentes de toda a chave primária.

Ministra = {Professor, Sigla, Livro} OK!!

Turma = {Número, Sigla, Sala, No.Horas} Não!!

Número, Sigla → Sala  
Sigla → Livro, No.Horas  
Professor → Sigla



# Normalização

---

## Como Normalizar Relações para 2<sup>a</sup>. Forma Normal

- Uma relação que não esteja na 2<sup>a</sup>. Forma Normal pode ser colocada nela.
  - ▶ Verificam-se os grupos de atributos que dependem da mesma parte da chave.
  - ▶ Retiram-se da relação todos os atributos de um desses grupos.
  - ▶ Cria-se uma nova relação, que tem esse grupo como atributos não chave, e os atributos que determinam esse grupo como chave.

# Normalização

Como Normalizar Relações para 2<sup>a</sup>. Forma Normal

Numero, Sigla  $\rightarrow$  Sala  
Sigla  $\rightarrow$  No.Horas

Turma = {Numero, Sigla, Sala, No.Horas}

O grupo de atributos que  
faz parte da chave e  
determina o outro grupo.

Um grupo de atributos  
que depende de parte da  
chave.

# Normalização

Como Normalizar Relações para 2<sup>a</sup>. Forma Normal

Numero, Sigla  $\rightarrow$  Sala  
Sigla  $\rightarrow$  No.Horas

Turma = {Numero, Sigla, Sala}

O grupo de atributos que faz parte da chave e determina o outro grupo.

Um grupo de atributos que depende de parte da chave.

Disciplina = {Sigla, No.Horas}



The diagram consists of two yellow arrows. One arrow originates from the text 'O grupo de atributos que faz parte da chave e determina o outro grupo.' and points to the underlined attribute 'Sigla' in the 'Disciplina' set. The other arrow originates from the text 'Um grupo de atributos que depende de parte da chave.' and points to the attribute 'No.Horas' in the 'Disciplina' set.

# Normalização

---

## Controlando a Consistência

### ■ Atenção:

- ▶ A normalização da relação é feita na grande maioria das vezes decompondo-se uma relação em duas ou mais outras.
- ▶ No entanto, é frequente que nas consultas seja necessário recompor a relação original.
- ▶ Isso não causa nenhum transtorno, uma vez que a não normalização de relações causa anomalias de atualização, mas não de consultas.
- ▶ Assim, recompor uma relação para uma consulta não traz anomalias, e para a atualização trabalha-se com as relações decompostas, com o que evitam-se anomalias e inconsistências .

# Normalização

---

## Controlando a Consistência

- Portanto, Atenção:

- ▶ A normalização da relação é feita na grande maioria das vezes decompondo-se uma relação em duas ou mais outras.

- Mas:

- ▶ A decomposição deve ser feita com cuidado, para evitar perda de informação na recomposição.

# Normalização

---

## Decomposição sem Perda na Junção

- A decomposição deve ser feita de maneira que quando recompõe-se a relação original, apenas e exatamente as tuplas existentes na relação original são re-obtidas.
- A decomposição baseada nas dependências funcionais não causam perdas de junção, portanto a normalização para as formas normais baseadas em dependências funcionais estão livres desse problema.

# Normalização

---

## Normalização de Relações

- A 2<sup>a</sup>. Forma Normal evita:
  - ▶ Inconsistência devida a duplicidade de informação.
  - ▶ Perda de dados em operações de remoção/alteração na relação

# Normalização

## Controlando a Consistência

Inconsistência devida a duplicidade de informações

Sigla → N<sup>o</sup>. Horas

Sigla, Número → Horário

Turma = {Sigla, Número, Horário, No. Horas}

→ SEL122, 1, 10:00, 2 } ✓  
→ SEL122, 2, 14:00, 2 } ✓  
→ SEL189, 1, 8:00, 3 } ✗  
→ SEL189, 2, 15:00, 4 } ✗  
SEL134, 1, 16:00, 2



# Normalização

## Controlando a Consistência

### Perda de Dados Devida a Alterações na Relação

Sigla → N<sup>o</sup>. Horas

Sigla, Número → Horário

Turma = {Sigla, Número, Horário, No. Horas}

SEL122, 1, 10:00, 2

SEL122, 2, 14:00, 2

SEL189, 1, 8:00, 3

SEL189, 2, 15:00, 4

⇒ SEL134, 1, 16:00, 2

Se não houver uma turma da disciplina neste semestre, eu perco a informação de quantas horas tem esta disciplina.

# Normalização

---

## Controlando a Consistência

Turma = {Sigla, Número, Horário}

SEL122, 1, 10:00

SEL122, 2, 14:00

SEL189, 1, 8:00

SEL189, 2, 15:00

SEL134, 1, 16:00

Turma = {Sigla, N°·Horas}

SEL122, 2

SEL189, 3

SEL134, 2

# Normalização

---

## Decomposição sem Perda na Junção

- No entanto, se a decomposição não for efetuada levando-se em conta estritamente as dependências funcionais, pode-se obter relações com perda de informações quando for feita uma junção para recompor a relação original:

# Normalização

## Decomposição COM Perda na Junção

Sigla → N<sup>o</sup> Horas  
Sigla, Número → Horário

### Decompondo-se:

Turma = {Sigla, Número, Horário, No. Horas}

SEL122, 1, 10:00, 2

SEL122, 2, 14:00, 2

SEL189, 1, 8:00, 3

SEL189, 2, 15:00, 4

SEL134, 1, 14:00, 3

Turma = {Sigla, Número, Horário}

SEL122, 1, 10:00

SEL122, 2, 14:00

SEL189, 1, 8:00

SEL189, 2, 15:00

SEL134, 1, 14:00

NHoras = {Horário, No.Horas}

10:00, 2

14:00, 2

8:00, 3

15:00, 4

14:00, 3

# Normalização

## Decomposição COM Perda na Junção

Turma = {Sigla, Número, Horário}    NHoras = {Horário, No.Horas}

Turma ⋈ NHoras	SEL122, 1, 10:00	=	10:00, 2
	SEL122, 2, 14:00		14:00, 2
	SEL189, 1, 8:00		8:00, 3
	SEL189, 2, 15:00		15:00, 4
	SEL134, 1, 14:00		14:00, 3

Turma = {Sigla, Número, Horário, No. Horas}

SEL122, 1, 10:00, 2  
SEL122, 2, 14:00, 2  
SEL189, 1, 8:00, 3  
SEL189, 2, 15:00, 4  
SEL134, 1, 14:00, 3  
SEL134, 1, 14:00, 2  
SEL122, 2, 14:00, 3

**Tupla  
Espúrias**

# Normalização

## Normalização de Relações para a 3<sup>a</sup>. Forma Normal

- Uma relação esta na 3<sup>a</sup>. Forma Normal quando:
  - ▶ Esta na 1<sup>a</sup>. Forma Normal;
  - ▶ Todos os atributos que não participam da chave primária são dependentes não transitivos de toda a chave primária.

Número, Sigla → Sala, Horário

Professor → Sigla

Sigla → Livro

Turma = {Sigla, Número, Sala, Horário}

OK!!

Sigla = {Professor, Sigla, Livro} Não!

# Normalizacao

## Como Normalizar Relações para 3<sup>a</sup>. Forma Normal

- Uma relação que não esteja na 3<sup>a</sup>. Forma Normal pode ser colocada nela.
  - ▶ Verifica-se um grupo de atributos que depende não diretamente da chave.
  - ▶ Retira-se da relação esse grupo de atributos.
  - ▶ Cria-se uma nova relação que contém esse grupo de atributos, e inclui-se nela como chave os atributos dos quais esse grupo depende diretamente.
  - ▶ Repetem-se esses passos até que todos os atributos restante na relação original dependam diretamente de toda sua chave.

# Normalização

---

## Exemplo

Turma = {Sigla, Número, Sala, Prédio}

Número, Sigla → Sala

Sala → Prédio

Turma = {Sigla, Número, Sala}

Salas = {Sala, Prédio}





# Normalização

---

## Normalizações de relações

- Da mesma maneira que a 2<sup>a</sup> Forma Normal a 3<sup>a</sup> Forma Normal evita:
  - ▶ Incosistência devida a duplicidade de informações:
  - ▶ Perda de dados em operações de remoção/alterações na relação.

# Normalização

Inconsistência devida a duplicidade de informações

Número, Sigla → Sala  
Sala → Prédio

Turma = {Sigla, Número, Sala, Prédio}

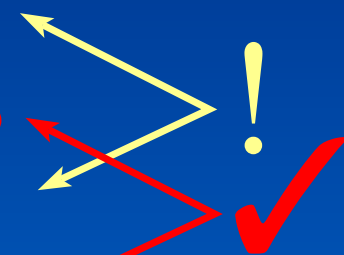
SFI102, 1, 3, E1

SFI102, 2, 4, E1

SFI104, 1, 12, C2

SFI155, 1, 4, C2

SFI155, 1, 12, C2



# Normalização

Perda de Dados devida a alterações na relação

Número, Sigla → Sala  
Sala → Prédio

Turma = {Sigla, Número, Sala, Prédio}

Se não houver uma aula nesta sala neste semestre, eu perco a informação de em qual prédio fica a sala.

SFI102, 1, 3, E1  
SFI102, 2, 4, E1  
SFI104, 1, 12, C2  
SFI155, 1, 4, C2  
SFI155, 1, 12, C2

# Normalização

Perda de Dados devida a alterações na relação

Número, Sigla  $\rightarrow$  Sala

Sala  $\rightarrow$  Prédio

~~Turma = {Sigla, Número, Sala, Prédio}~~

Turma = {Sigla, Número, Sala}   Salas = {Sala, Prédio}

SFI102, 1, 3

3, E1

SFI102, 2, 4

4, E1

SFI104, 1, 12

12, C2

SFI155, 1, 4

SFI155, 1, 12

# Normalização

---

## Forma Normal de Boice - Codd (BCNF)

- BCNF é uma extensão da 3<sup>a</sup> Forma Normal, onde estende-se a regra para cada uma das chaves candidatas da relação:
- Uma relação está em BCNF normal quando:
  - ▶ Esta na 1<sup>a</sup> Forma normal;
  - ▶ Para todas as chaves candidatas vale a regra de que: todos os atributos que não participam dessa chave são dependentes diretos de toda essa chave .

# Normalização

---

## Forma Normal de Boice - Codd (BCNF)

- Para relações que tenham apenas uma chave candidata, a BCNF reduz-se à 3<sup>a</sup> Forma Normal.
- Quando existe mais de uma chave candidata, a BCNF deve verificar a total dependência dos atributos primos (atributos pertencente a uma chave candidata) de uma chave diretamente com as chave.
- A normalização de uma relação que não está em BCNF é empreendida da mesma maneira que para a 3<sup>a</sup> Forma Normal, verificando-se cada chave candidata da relação.

# Normalização

## Exemplo

Sala, Horário → Professor, Sigla, N#Alunos  
Horário, Professor → Sala, Sigla

Aloca = {Sala, Horário, Professor, Sigla, N#Alunos}

Para a BCNF, deve-se considerar todas as chaves: {Sala, Horário} e {Horário, Professor}

Nesse caso, a relação “Aloca” não está em BCNF, pois a chave {Horário, Professor} determina N#Alunos apenas (pseudo-) transitivamente através de Sala.



# Normalização

---

## Considerações sobre à 2<sup>a</sup>.FN, 3<sup>a</sup>.FN e BCNF

- A normalização é feita uma relação por vez.
- O processo de normalização vai “quebrando” as relações, e portanto criando outras.
- Quando uma relação está em uma determinada forma normal, o processo de normaliza-lá para uma forma mais rígida gera relações que mantêm-se na forma normal menos rígida em que a relação original encontrava-se.



# Normalização

## Considerações sobre à 2<sup>a</sup>FN, 3<sup>a</sup>FN e BCNF

- Para a determinação de dependência de atributos a partir das chaves, consideram-se conjuntos de atributos que dependem-se mutuamente como que formando um único grupo.

Aluno = {Nome, N#USP, CIC, Idade}

Nome  $\rightarrow$  N#USP, CIC, Idade

N#USP  $\rightarrow$  Nome, CIC, Idade

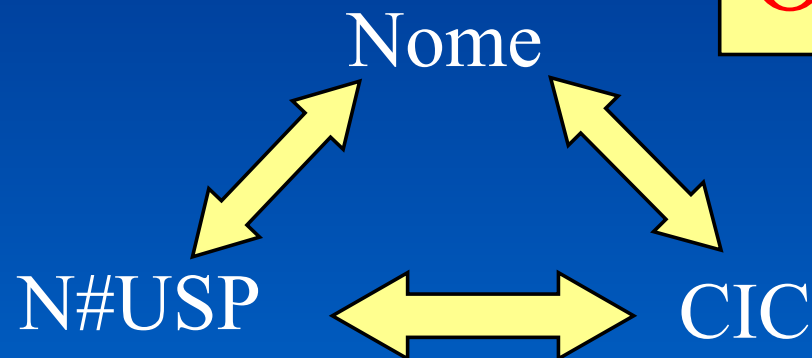
CIC  $\rightarrow$  N#USP, Nome, Idade

# Normalização

Considerações sobre à 2<sup>a</sup>FN, 3<sup>a</sup>FN e BCNF

- Para a determinação de dependência de atributos a partir das chaves, consideram-se conjuntos de atributos que dependem-se mutuamente como formando um único grupo.

Aluno = {Nome, N#USP, CIC, Idade}



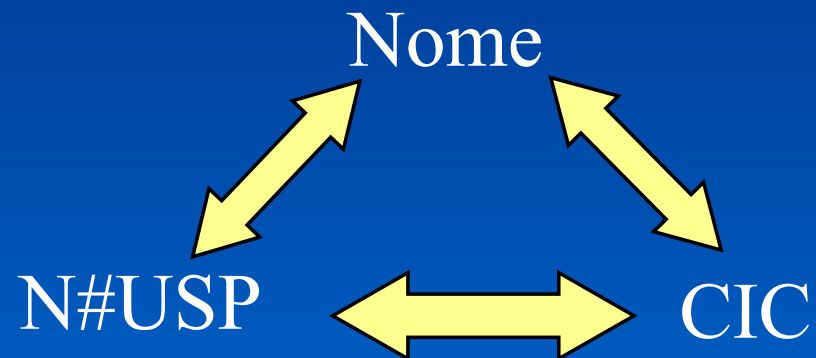
**OK!!**

# Normalização

Considerações sobre à 2<sup>a</sup>FN, 3<sup>a</sup>FN e BCNF

- Um grupo de atributos assim definem uma Relação de Equivalência sobre o operador determina (  $\rightarrow$  ).
- Apesar disso, esse encadeamento de dependências é chamado “Dependência Cíclica” .

Aluno = {Nome, N#USP, CIC, Idade}



# Normalização

Considerações sobre à 2<sup>a</sup>FN, 3<sup>a</sup>FN e BCNF

- Também não consideram-se para a avaliação das FN 2<sup>a</sup>, 3<sup>a</sup> e BC as chamadas “Dependências Triviais”.
- Dependências Triviais são aquelas que ocorrem em todas as Relações.

$$R \rightarrow A, \forall A \subseteq R$$

# Normalização

---

## Normalização de Relações

- A normalização de relações através de dependências funcionais é apenas uma das maneiras, embora a mais importante, de evitar inconsistências em relações.
- Uma outra maneira, advém de uma variação das dependências funcionais, chamadas dependência multivalorada, ou multi-dependência funcional.

# Normalização

---

## Multi-Dependência Funcional

- A dependência funcional significa que o valor de um atributo pode ser determinado a partir de outras.
- A multi-dependência funcional caracteriza o fato de que, embora um conjunto de atributos não possa determinar o valor de outro atributo, ainda assim esse conjunto consegue restringir os valores possíveis para aquele atributo.
- Tal como ocorre com a dependência funcional, a multi-dependência não pode ser inferida pelo SGBD, e portanto também deve ser informado pelo programador ao Gerenciador de Base de Dados.

# Normalização

## Multi-Dependência Funcional

- Dada uma relação  $R = \{A_1, A_2, \dots, A_n\}$  se um conjunto de atributos  $A \subset R$  restringe os valores possíveis para os atributos de outro conjunto  $B \subset R$ , então diz-se que  $B$  é **Multi-dependente** de  $A$ , ou que  $A$  **Multi-determina**  $B$ , e nota-se:  $A \twoheadrightarrow B$ .
- A multi-dependência funcional ocorre quando na relação existem atributos multivalorados, que por força da 1<sup>a</sup>FN precisam ser desmembrados em múltiplas ocorrências de tupla que contém apenas um dos valores para aquele atributo.

# Normalização

## Multi-Dependência Funcional

- Suponha que exista na base de dados uma relação que indica para cada professor, qual seu nome, disciplina que ele pode lecionar e o nome dos alunos que ele orienta.

Prof. = {Nome, Matéria, Orientando}  
{<Carlos, S.O., Mário>,  
<Carlos, S.O., Ana>,  
<Alzira, B.D., Paulo>,  
<Alzira, B.D., Sônia>,  
<Alzira, E.S., Paulo>,  
<Alzira, E.S., Sônia>}



# Normalização

---

## Multi-Dependência Funcional

- Esta relação está em BCNF, até porque todos os atributos da relação são necessários para compor a única chave da relação. No entanto existe uma anomalia que pode causar problemas nas operações de modificação desta relação.
- Tanto *Matéria* quanto *Orientando* são atributos que não dependem funcionalmente do nome, cada professor pode orientar mais de um aluno, bem como ministrar mais de uma matéria. Por outro lado, mais de um professor pode ministrar a mesma matéria, e um aluno pode ter co-orientadores.

# Normalização

---

## Multi-Dependência Funcional

- No entanto, como *Matéria* e *Orientando* são atributos multi-valorados para um dado professor, são também atributos multi-dependentes de *Nome*:
  - $\text{Nome} \twoheadrightarrow \text{Matéria}, \text{Orientando}$
- Se Alzira aceitar mais um orientando, ela continua podendo ministrar suas mesmas matérias, portanto devem ser inseridos tuplas para cada disciplina que ela ministra juntamente com o nome do novo orientando.

# Normalização

## Multi-Dependência Funcional

Prof. = {Nome, Matéria, Orientando}

<Carlos, S.O., Mário>,  
<Carlos, S.O., Ana>,  
<Alzira, B.D., Paulo>,  
<Alzira, B.D., Sônia>,  
<Alzira, E.S., Paulo>,  
<Alzira, E.S., Sônia>  
<Alzira, B.D., Pedro>,  
<Alzira, E.S., Pedro>}

Na realidade, sempre que dois conjuntos de atributos multivalorados independentes ocorrerem na mesma relação, será necessário repetir todos os valores de cada um dos conjunto de atributos para cada valor possível do outro conjunto.

# Normalização

## Multi-Dependência Funcional

- Essa situação de simetria ocorre sempre. Assim, sempre que houver uma Multi-dependência funcional  $A \twoheadrightarrow B$  numa relação  $R$ ,  $A \subset R$ ,  $B \subset R$ , e um conjunto  $C = R - A - B$ , então também ocorrerá a multi-dependência  $A \twoheadrightarrow C$ .
- Assim,  $A \twoheadrightarrow B$  significa também que  $A \twoheadrightarrow C$  e pode-se escrever igualmente que  $A \twoheadrightarrow B/C$ .
- Teoricamente, pode-se dizer que a dependência Funcional (DF) é um caso particular da Multi-Dependência Funcional (MDF), aonde o número de valores que podem ser determinadas é exatamente 1.

# Normalização

---

## Multi-Dependência Funcional

- Da mesma maneira que nas dependências funcionais, as Multi-dependências funcionais devem ser especificadas pelo projetista, e deve-se conhecer o conjunto completo das multi-dependências funcionais existentes.
- Para isso, existe um conjunto de regras de inferência que permitem obter outras MDF a partir de um conjunto inicial.
- As regras de inferência seguintes constituem um conjunto completo que envolvem tanto a dependência funcional quanto a multi-dependência funcional.

# Normalização

---

## Multi-Dependência Funcional

- As três regras de derivações fundamentais referentes apenas à Dependências funcionais são as seguintes:
  - ▶ Reflexiva, Se  $A \subset B \Rightarrow B \rightarrow A$
  - ▶ Aumentativa, Se  $A \rightarrow B \Rightarrow AC \rightarrow B$
  - ▶ Transitiva, Se  $A \rightarrow B, B \rightarrow C \Rightarrow A \rightarrow C$

# Normalização

## Multi-Dependência Funcional

- Existem mais 3 regras Fundamentais referentes a Multi-Dependências Funcionais:
  - ▶ Complemento (MDF), Se  $A \twoheadrightarrow B \Rightarrow A \twoheadrightarrow (R - (A \cup B))$
  - ▶ Aumentativa (MDF), Se  $A \twoheadrightarrow B$  e  $C \subseteq D \Rightarrow AD \twoheadrightarrow BC$
  - ▶ Transitiva (MDF), Se  $A \twoheadrightarrow B$  e  $B \twoheadrightarrow C \Rightarrow A \twoheadrightarrow (C - B)$
- Existem mais 2 Regras Fundamentais que vinculam Dependências com Multi-Dependências Funcionais:
  - ▶ Duplicação (DF  $\rightarrow$  MDF), Se  $A \rightarrow B \Rightarrow A \twoheadrightarrow B$
  - ▶ Coalescência, Se  $A \twoheadrightarrow B$  e  $\exists C \mid C \cap B = \emptyset, C \rightarrow D, \text{ e } D \subseteq B \Rightarrow A \rightarrow D$

# Normalização

## Multi-Dependência Funcional

- Esse conjunto de 8 Regras é completo e suficiente para se obter todas as MDF possíveis a partir de um conjunto inicial.
- Além disso, a regra seguinte é frequentemente útil para obterem-se as diversas Multi-dependências funcionais derivadas mais rapidamente, embora todas possam ser obtidas a partir desse conjunto fundamental de 8 regras:
  - ▶ Se  $A \twoheadrightarrow B$  e  $A \twoheadrightarrow C \Rightarrow$ 
    - $A \twoheadrightarrow BC$
    - $A \twoheadrightarrow B \cap C$
    - $A \twoheadrightarrow B - C$
    - $A \twoheadrightarrow C - B$



# Normalização

## Quarta Forma Normal

- Sempre que houver a ocorrência de um atributo multi-valorado ocorrerá a multi-dependência funcional. Se houver duas ou mais dependências multi-funcionais independentes entre si na mesma relação então ocorrerão anomalias de atualização na relação. Isso deve ser evitado.
- Assim, o que procura-se é a chamada multi-dependência funcional não trivial.
- A multi-dependência funcional trivial é definida como:
  - Numa relação  $R$ , a multi-dependência  $A \twoheadrightarrow B$  é dita trivial: se  $B \subset A$  ou se  $A \cup B = R$ .

# Normalização

---

## Quarta Forma Normal

- Em outras palavras, de uma maneira mais intuitiva, pode-se dizer que uma multi-dependência funcional não trivial ocorre sempre que houver mais do que um atributo proveniente de um atributo multi-valorado na mesma relação.
- Com esses conceitos é possível agora definir-se a Quarta Forma Normal.

# Normalização

## Quarta Forma Normal

- Uma relação está na 4<sup>a</sup>. Forma Normal quando:
  - ▶ Dado um conjunto completo de Multi-dependência funcionais não triviais para essa relação, para todas as MDFNT  $A \twoheadrightarrow B$  desse conjunto então  $A$  é uma Superchave da Relação.

Prof. = {Nome, Matéria, Orientando}

Nome  $\twoheadrightarrow$  Matéria, Orientando

Nome não é superchave de Prof., portanto  
Prof. não está na 4<sup>a</sup>. Forma Normal.

# Normalização

## Quarta Forma Normal

- Para cada MDFNT  $A \twoheadrightarrow B$  que ocorre na relação  $R$ , deve-se substituir a relação pelas relações:
  - $R - B$  e  $A \cup B$

Prof. = {Nome, Matéria, Orientando}

Nome  $\twoheadrightarrow$  Matéria, Orientando

Ministra = {Nome, Matéria}

Orienta = {Nome, Orientando}

# Normalização

---

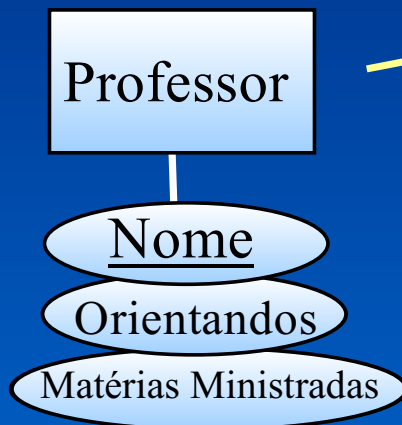
## Quarta Forma Normal

- A 4a. Forma Normal evita:
  - ▶ Inconsistências devidas à inclusão de uma nova tupla que tem valores diferentes das diversas ocorrências de um outro atributo Multi-valorado.
  - ▶ Inconsistências em operações de remoção de tuplas, sendo que o produto cartesiano dos atributos multivalorados da relação possuem diferentes valores de um dos atributos em comparação com os valores do outro atributo.

# Normalização

## Multi-Dependência Funcional

- Numa modelagem segundo o ME-R, quando um Conjunto de Entidades ou de Relacionamentos possui um atributo multivalorado, no mapeamento para o Modelo Relacional esses atributos, a princípio, constituem cada qual uma nova relação.

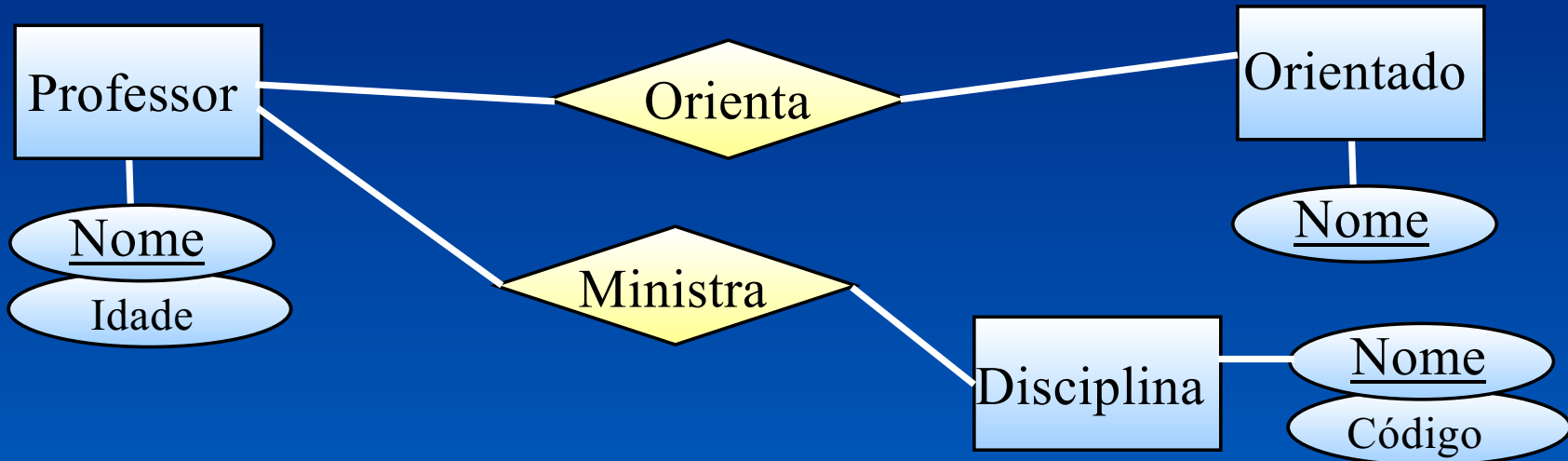


Orientadores = {Nome, Orientando}  
Ministra = {Nome, Matéria}

# Normalização

## Multi-Dependência Funcional

- Os mesmos atributos multivalorados podem corresponder a conjunto de entidades distintos, se houver necessidade de se armazenar mais dados a respeito de cada um deles. Nesse caso, os atributos multi-valorados aparecem como conjunto de relacionamentos de cardinalidade M:N.



# Normalização

## Multi-Dependência Funcional

- No entanto, o fato dos atributos não precisarem de mais informações além dos seus próprios valores atômicos, permite adotar a forma de atributos multi-valorados.
- Se a modelagem for feita originalmente no MRel, pode-se modelar mais de um atributo multi-valorado na mesma relação. Isso representa a mesma informação, mas possibilita que ocorram anomalias de atualização.

Orientadores = {Nome, Orientando, Matéria}



# Normalização

---

## Multi-Dependência Funcional

- Esse é o erro de modelagem que a 4a. Forma Normal procura evitar.
- A modelagem mapeada a partir do ME-R, se bem feita, evita essa anomalia.

Orientadores = {Nome, Orientando}

Ministra = {Nome, Matéria}

# Normalização

---

## Quarta Forma Normal

### ■ Características marcantes:

- ▶ Uma relação não pode conter um atributo multi-valorado que não é superchave, pois para uma mesma chave, deveriam haver diversas tuplas para contemplar os diversos valores desse atributo.
- ▶ Assim, todos os atributos que sejam multivalorados devem fazer parte da chave.
- ▶ Portanto, somente tem sentido verificar-se a multi-dependência funcional em relações que tenham 3 ou mais atributos como parte da chave.

# Normalização

---

## Quinta Forma Normal

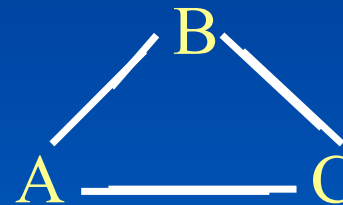
- Existe uma outra forma normal, denominada Quinta Forma Normal, que também advem das dependências multi-valoradas que ocorrem entre os atributos de uma relação.
- A verificação da 5<sup>a</sup>. FN somente precisa ser empreendido em relações que tenham 3 ou mais atributos como parte da chave.

# Normalização

## Quinta Forma Normal

- A 5<sup>a</sup>. FN ocorre quando existe uma multi-dependência cíclica entre pelo menos 3 conjunto de atributos da chave da relação.
- A 5<sup>a</sup>. FN é importante do ponto de vista que a decomposição de uma relação em um conjunto de outras relações (devido principalmente a necessidades de normalização), quando envolve multi-dependências cíclicas implicam na perda de informações devido à dependência de junção.

$R = \{A, B, C\}$



# Normalização

## Quinta Forma Normal

- Ao contrário das formas normais apoiadas em dependências funcionais e da 4<sup>a</sup>. FN, a 5<sup>a</sup>. FN não analisa apenas uma única relação e os seus atributos.
- A 5<sup>a</sup>. FN impõe que se avalie todo o conjunto de multi-dependências funcionais, para verificar se não existem dois conjuntos de atributos  $A$  e  $B$  da relação  $R$  em questão, tal que  $A \subset R$  e  $B \subset R$ , e existe alguma multi-dependência da forma  $A \twoheadrightarrow B$  que participa de algum ciclo de multi-dependências funcionais com 3 ou mais conjunto de atributos.
- Com esses conceitos podemos definir a 5<sup>a</sup>. FN:

# Normalização

---

## Quinta Forma Normal

- Uma relação  $R$  está na 5<sup>a</sup>. FN quando:
  - ▶ Dado o conjunto de todas as multi-dependências funcionais de uma base de dados, a relação  $R$ :
    - Ou não tem nenhum conjunto de atributos  $A$  que pertença a um ciclo de MDF com ao menos 3 conjuntos de atributos.
    - Ou os conjuntos de atributos que pertençam a um ciclo estão todos nessa relação.

# Normalização

---

## Quinta Forma Normal

- Uma relação não estar na 5<sup>a</sup>. FN significa que, quando duas relações têm atributos envolvidos em um mesmo ciclo de MDF sofrerem uma operação de junção serão geradas tuplas espúrias.
- Essas tuplas espúrias podem ser identificadas apenas através da intersecção com todas as relações obteníveis através de junções dois a dois de relações que têm atributos envolvidos nesse mesmo ciclo de MDF.

# Normalização

## Quinta Forma Normal

- A 5<sup>a</sup>. FN impõe que os atributos envolvidos nos conjuntos que constituem um mesmo ciclo de MDF estejam todos na mesma relação. Assim não se torna necessária a operação de junção e consequentemente não criam-se tuplas espúrias.
- Como normalizar para a 5<sup>a</sup>. FN
  - Dessa maneira, normalizar-se um conjunto de relações para a 5<sup>a</sup>. FN significa compor uma nova relação que seja a união de todas relações que tenham alguma MDF de um mesmo ciclo.



# Normalização

---

## Quinta Forma Normal

- Note-se que a 5<sup>a</sup>. FN atua de maneira contrária a 4<sup>a</sup>. FN, pois uma relação que não esteja em 4<sup>a</sup>. FN terá que ser decomposta para ser normalizada, em um conjunto de relações que poderão não ficar em 5<sup>a</sup>. FN.
- Por outro lado, uma relação que não esteja em 5<sup>a</sup>. FN terá que ser recomposta com as outras relações que têm atributos envolvidos no mesmo ciclo de MDF, “desnormalizando” dessa maneira essas relações quanto à 4<sup>a</sup>. FN.

# Normalização

## Quinta Forma Normal

- Note-se que é possível uma relação estar em 4<sup>a</sup>. FN e em 5<sup>a</sup>. FN se não houverem ciclos de MDF nos atributos da base (tomada como um todo).
- Basta que não exista nenhum ciclo de MDF envolvendo nenhum conjunto de atributos que constitua essa relação.
- Se existirem esses ciclos, não será possível normalizar-se essa relação para 4<sup>a</sup>. FN e 5<sup>a</sup>. FN simultaneamente.
- Nesse caso, é preferível manter-se a relação em 5<sup>a</sup>. FN, pois é mais difícil para um aplicativo garantir a 5<sup>a</sup>. FN do que a 4<sup>a</sup>.FN.

# Normalização

## Quinta Forma Normal

- Exemplo:
  - Considere-se que numa escola cada professor pode ministrar diversas disciplinas, mas não todas.
  - Além disso, eventualmente grupos de professores unem-se para fazer uma nova apostila, de maneira que cada apostila é de autoria de um número restrito de professores.
  - Outra restrição que deve ser armazenada na base é o fato de que cada disciplina sempre tem uma ou mais apostilas que podem ser usadas para acompanhamento.
- Nesse caso, as multi-dependências identificadas são:

Professor ->> Disciplina, Apostila

Apostila ->> Professor

Disciplina ->> Apostila

# Normalização

## Quinta Forma Normal

Atividades = {Prof., Apost., Discipl.}

$\left. \begin{array}{l} \langle \text{Zé}, \text{SO1}, \text{SO} \rangle, \\ \langle \text{Zé}, \text{BD1}, \text{BD} \rangle, \\ \langle \text{Mané}, \text{BD1}, \text{BD} \rangle, \\ \langle \text{Mané}, \text{ED1}, \text{ED} \rangle, \\ \langle \text{Mané}, \text{BD2}, \text{BD} \rangle, \\ \langle \text{Zé}, \text{BD2}, \text{BD} \rangle, \end{array} \right\} R1 \boxtimes R3$

$R1 \boxtimes R2 \longrightarrow 5 \text{ tuplas espúrias!}$

$R3 \boxtimes R2 \longrightarrow \text{OK!!}$

$R1 = \{ \text{Prof.}, \text{Discipl.} \}$   
 $\langle \text{Zé}, \text{SO} \rangle,$   
 $\langle \text{Zé}, \text{BD} \rangle,$   
 $\langle \text{Mané}, \text{BD} \rangle,$   
 $\langle \text{Mané}, \text{ED} \rangle$

$R2 = \{ \text{Prof.}, \text{Apost.} \}$   
 $\langle \text{Zé}, \text{SO1} \rangle,$   
 $\langle \text{Zé}, \text{BD1} \rangle,$   
 $\langle \text{Mané}, \text{BD1} \rangle,$   
 $\langle \text{Mané}, \text{BD2} \rangle,$   
 $\langle \text{Mané}, \text{ED1} \rangle$

$R3 = \{ \text{Apost.}, \text{Discipl.} \}$   
 $\langle \text{SO1}, \text{SO} \rangle,$   
 $\langle \text{BD1}, \text{BD} \rangle,$   
 $\langle \text{BD2}, \text{BD} \rangle,$   
 $\langle \text{ED1}, \text{ED} \rangle$

# Normalização

---

## Considerações quanto à Normalização

- Existem diversas outras formas normais que podem ser definidas. Como por exemplo restrições sobre unicidade de chaves, restrições sobre valores de atributos a domínios ou expressões de cálculo intra ou inter-tuplas, etc.
- No entanto, as formas normais apresentadas são ao mesmo tempo as mais genéricas e as que mais impacto causam sobre o projeto de dados de uma aplicação.
- Assim, aqui considera-se apenas as FN já apresentadas. Serão feitas agora algumas considerações sobre seu emprego.

# Normalização

## Considerações quanto à Normalização

- As formas normais apoiadas em dependências funcionais (2<sup>a</sup> FN, 3<sup>a</sup> FN e BCNF) podem ser consideradas para cada relação. Uma base de dados será considerada normalizada para uma dessas formas quando todas as suas relações se apresentarem nessa forma.
- As demais formas normais devem levar sempre em consideração a base de dados como um todo. Por exemplo, apenas estará em 1<sup>a</sup> FN as relações cujos atributos não aparecem “partidos” em nenhuma outra relação da base. Nesse caso considera-se que cada relação esteja nessa determinada FN se a base toda a estiver.

# Normalização

---

## Considerações quanto à Normalização

- A normalização para as FN apoiadas em DF sempre se atinge através da separação dos atributos “faltosos” em duas ou mais relações. Isso aumenta não apenas o número de relações, mas também obriga que a recuperação de informações obrigue a junção de diversas relações.
  - ▶ Normalizar evita introduzir inconsistências quando se alteram relações;
  - ▶ Porém obriga a execução de custosas operações de junção para a consulta de informações.

# Normalização

---

## Considerações quanto à Normalização

- A decisão de normalizar-se ou não uma relação é um compromisso entre se garantir a eliminação de inconsistências na base, e eficiência de acesso.
- A normalização para formas apoiadas em DF evita inconsistências, usando para isso a própria construção da base. Se a mesma consistência puder ser garantida pelos aplicativos, a normalização correspondente não precisa ser realizada, ganhando-se tempo nas consultas.
- Igualmente, se a consistência não for um fator fundamental, pode-se abrir mão da normalização.



# Normalização

## Considerações quanto à Normalização

Nome  $\rightarrow$  Rua, Número, Cidade, CEP, Fone  
Rua, Cidade  $\rightarrow$  CEP



Aluno = {Nome, Rua, Número, Cidade, Fone}  
CEPs = {Rua, Cidade, CEP}



```
SELECT *  
FROM Aluno, CEPs  
WHERE CEPs. Rua = Aluno.Rua AND  
       CEPs.Cidade = Aluno.Cidade
```

# Normalização

## Considerações quanto à Normalização

- As formas normais apoiadas em Multi-Dependências funcionais ( $4^{\text{a}}$  FN e  $5^{\text{a}}$  FN) devem se consideradas tanto localmente em cada relação ( $4^{\text{a}}$  FN), quanto sobre toda a base de dados ( $5^{\text{a}}$  FN).
- O impacto dessas FN num projeto de dados é bem menos do que as FN apoiadas em DF. EM primeiro lugar, as MDFNT são mais raras, e em geral já levadas em conta quando são “Montadas” as relações.
- Em segundo lugar, é uma tarefa difícil a localização de muitas das multi-dependências funcionais, que ficam ocultas pelo fato de um conjunto de atributos não ser capaz de atuar como chave de acesso a outros atributos.

# Banco de Dados

---

Modelo Relacional  
– Normalização –

Caetano Traina Júnior

FIM